ASDV 2620, Lab
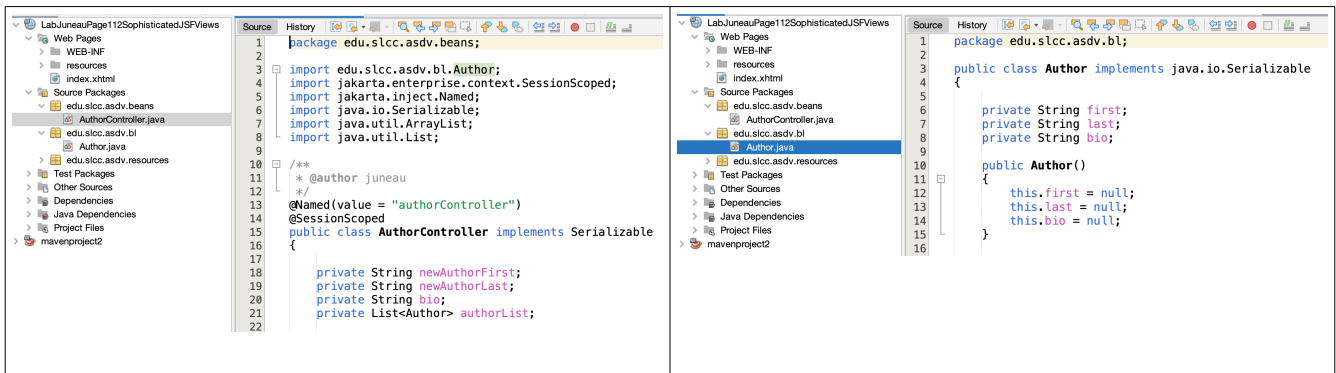**Juneau-Textbook page 112**
Building Sophisticated JSF Views with Components

Lab Part 1

1. Create a Maven Web Application with JSF (Jakarta 9, JSF 3.0). Your index.xhtm is shown below.

```
1   <?xml version='1.0' encoding='UTF-8' ?>
2   <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
3   <html xmlns="http://www.w3.org/1999/xhtml"
4         xmlns:h="http://xmlns.jcp.org/jsf/html"
5         xmlns:f="http://xmlns.jcp.org/jsf/core">
6       <h:head>
7           <meta http-equiv="Content-Type" content="text/html; charset=UTF-8"/>
8           <title>Building Sophisticated JSF Views with Components</title>
9       </h:head>
10      <h:body>
11          <h:form id="componentForm">
12              <h1>JSF Components, Creating a Sophisticated Page</h1>
13              <p>
14                  The view for this page is made up entirely of JSF standard components.
15                  <br/>As you can see, there are many useful components bundled with JSF out of the box.
16              </p>
17              <p>Book Recommendation: Java 7 Recipes
18                  <br/>
19                  <h:graphicImage id="java7recipes" library="images" name="recipe.jpg"/>
20                  <br/>
21                  <p>
22                      Use the following form to add an author to the list.
23                  </p>
24                  <h:outputLabel for="newAuthorFirst" value="New Author First Name: "/>
25                  <h:inputText id="newAuthorFirst" value="#{authorController.newAuthorFirst}"/>
26                  <br/>
27                  <h:outputLabel for="newAuthorLast" value="New Author Last Name: "/>
28                  <h:inputText id="newAuthorLast" value="#{authorController.newAuthorLast}"/>
29                  <br/>
```

```
30                  <h:outputLabel for="bio" value="Bio:"/>
31                  <br/>
32                  <h:inputTextarea id="bio" cols="20" rows="5"
33                                   value="#{authorController.bio}"/>
34                  <br/>
35                  <br/>
36                  <h:commandButton id="addAuthor" action="#{authorController.addAuthor}"
37                                   value="Add Author"/>
38                  <br/>
39                  <br/>
40                  <h:dataTable id="authorTable" value="#{authorController.authorList}"
41                               var="author">
42                      <f:facet name="header">
43                          Java 7 Recipes Authors
44                      </f:facet>
45                      <h:column>
46                          <h:outputText id="authorName" value="#{author.first} #{author.last}"/>
47                      </h:column>
48                  </h:dataTable>
49                  <br/>
50                  <br/>
51              </p>
52          </h:form>
53      </h:body>
54  </html>
55
56
```

2. Create the CDI bean AuthorController and the pojo Author as shown below:


```
package edu.slcc.asdv.beans;

import edu.slcc.asdv.bl.Author;
import jakarta.enterprise.context.SessionScoped;
import jakarta.inject.Named;
import java.io.Serializable;
import java.util.ArrayList;
import java.util.List;

/**
 * @author juneau
 */
@Named(value = "authorController")
@SessionScoped
public class AuthorController implements Serializable
{

    private String newAuthorFirst;
    private String newAuthorLast;
    private String bio;
    private List<Author> authorList;
```

```
package edu.slcc.asdv.bl;

public class Author implements java.io.Serializable
{

    private String first;
    private String last;
    private String bio;

    public Author()
    {
        this.first = null;
        this.last = null;
        this.bio = null;
    }
```

3. Clean and build and run.

localhost:8080/LabJuneauPage112SophisticatedJSFViews/faces/index.xhtml

Use the following form to add an author to the list.

New Author First Name: John
New Author Last Name: Wayne
Bio:
Good man!

Add Author

Java 7 Recipes Authors
Josh Juneau
Carl Dea
Mark Beaty
John O'Conner
Freddy Guime
John Wayne

4. Understand the explanation given by the author in pages 119, 123

5. Submit zip plus HTML of CDI bean

Lab Part 2

6. Copy/paste and refactor your project into LabJuneauPage122Refined.

7. **A**dd a property file for all titles used in index.xhtml (i.e first name)

8. In the CDI bean leave just a place holder for the business logic that will use the methods of the business logic. Take the initialization in the constructor and pace it into a static method somewhere you decide, inside you business logic.

9. Submit zip, plus CDI bean, plus index.xhtml.